

# **Research project report: Modelling MAPK/ERK pathway using Ordinary Differential Equations**

Przemysław Pilipczuk

Abstract text

October 13, 2025

Preface text

# Contents

1. Abstract .....	4
2. Introduction .....	5
3. Methods .....	6
3.1. Ordinary Differential Equations .....	6
3.2. Modeling & Simulation pipeline .....	6
4. Results .....	7
5. Discussion .....	8
5.1. Time resolution trouble .....	8
6. Conclusions .....	9
7. Future work .....	10
8. Supplementary material .....	11
Bibliography .....	12

## 1. Abstract

The MAPK/ERK pathway serves a key role in determining fate of a cell from extracellular inputs. Different cell fates are linked to a dynamics of the final node in the cascade, ERK kinases. To understand how different dynamic patterns of their behavior arise, we attempt to mechanistically model the simplified version of this pathway using Ordinary Differential Equations. Using data from a fibroblast cell transfected with optoRTK constructs and stimulated with various patterns of light, we attempted to estimate model parameters based on these experiments and use cross-validation techniques to check for generalization. Resulting fits had problems with achieving several performance metrics that would signify an optimal solution.

## 2. Introduction

TODO

## 3. Methods

### 3.1. Ordinary Differential Equations

Ordinary differential equations is a mathematical framework for describing the change of one variable (dependant variable, here  $Y$ ) over another variable (independent variable, here  $X$ ). Its simplest formulation is an expression that equates some mathematical expression to a derivative of our variable in question over the independent variable.

$$\frac{dY}{dX} = \dots \quad (1)$$

A solution to a differential equation is usually understood as obtaining a  $Y(X)$  form, also called an *general solution*. General solutions can be obtained using an analytical solving process, the difficulty of which is heavily dependant on the specific problem being solved. An often encountered problem with analytical solution approach is a problem that contains complex, nonlinearly coupled equations, which do not yield easily to this method. An alternative approach to solving a system that has these characteristics is a numerical one. This method relies on a provided initial conditions and a `x-step` resolution to simulate a single trajectory within an ODE system by evaluating the equations sequentially at a consecutive `x-steps` away from the provided starting point. This method, while providing a weaker form of solution, can deal with harder problems, including those that describe complex, nonlinear systems.

### 3.2. Modeling & Simulation pipeline

Model definition was done in symbolic form using python and SymPy (TODO: reference them). This same tools was also used to lower the symbolic expression down to a numerical representation ( `lambdify` function). A constructed model consists of a set of differential equations and parameter values. An effort was made to make this pipeline model-and-experiment agnostic and have low friction for implementing new models. This effort led to an architecture where each new model needs only to fulfill a simple interface for defining its equations and parameters, and each new experiment needs to define its own pattern of input and a parsing function for the data it generated.

An interactive environment for simulating behavior and exploring perturbations to parameters was constructed using marimo notebooks.

Parameter estimation was done using tools from SciPy library, namely `minimize` and `solve_ivp` to optimize loss function and to solve a numerical system respectively.

## 4. Results

## 5. Discussion

### 5.1. Time resolution trouble

During implementation of the parameter estimation regime, a technical difficulty was encountered. The experiments had been replicated multiple times with different parameter values for light stimulation, usually differing in stimulation time of a pulse of light, with values ranging from 50ms to 2000ms. At the same time, data for those experiments was collected with a rate of one sample per minute. Such difference in time ranges between sampling and input means that our experimental data will never be able to differentiate between the light stimulation duration groups within an experiment just by looking at the data (we would need to increase the sampling rate 1200 times to be able to discern signals that lasted 50ms, the most fine-grained light signal in our experimental data). To circumvent this problem, two solutions were contemplated.

1. An interpolation algorithm could have been used to synthetically create data points of the sampling rate we desire. This forces our new data points to inherit our assumptions about how the data is interpolated, which could introduce a systematic bias (as example, for linearity).
2. A proxy metric for a total light energy of a light pulse for a datapoint is introduced instead of a “binary” baseline. This allows for more flexibility but at a cost of having to create a realistically-scaling function that works for all kinds of experiments and across different levels of magnitude of light stimulation duration.

After theoretical evaluation of pros and cons of both, a second solution strategy was picked, due to a better compatibility with the goal of having short training times.



## 6. Conclusions

## **7. Future work**

## **8. Supplementary material**

## **Bibliography**